What is claimed is:

1. A method of evaluating a dependency graph of a graphics creation process, comprising:

passing a function of a first dependency node to a second dependency node; and

evaluating the function as part of an evaluation of the second dependency node.

2. A method as recited in claim 1, wherein the function comprises a self evaluating data structure.

3. A method as recited in claim 2, wherein the function comprises a function having a defined set and type of inputs and outputs.

4. A method as recited in claim 2, wherein the structure comprises a function pointer.

5. A method as recited in claim 2, wherein the structure comprises a function calling method.

6. A method as recited in claim 2, wherein the evaluating comprises determining a type of a passed parameter.

7. A method as recited in claim 6, wherein the function

parameter types are identified dynamically as the dependency graph is executed.

8. A method as recited in claim 7, wherein the data structure contains information describing a set of input and output parameters the function accepts.

9. A method as recited in claim 8, wherein the information determines if function attribute types within the dependency graph are compatible.

10. A method as recited in claim 9, wherein the data structure comprises default values for all input and output parameters.

11. A method as recited in claim 1, further comprising mapping parameters of first and second functions of the first and second nodes.

12. A method as recited in claim 11, wherein said mapping comprises using an index.

13. A method as recited in claim 11, wherein the mapping defines a relationship where input parameters are ignored and output parameters are unmapped and take on default values.

14. A method as recited in claim 11, wherein parameter value and type are passed for the mapping.

15. A method as recited in claim 11, wherein the function data structure and value index are passed for the mapping.

16. A method as recited in claim 11, wherein the mapping comprises an index remapping and a matrix of data casting methods which will change one type of data into another.

17. A method of evaluating a dependency graph of a graphics creation process, comprising:

passing a function of a first dependency node to a second dependency node, the function comprising a self evaluating data structure

comprising a function calling method and containing information describing a set of input and output parameters the function accepts where the information determines if function attribute types within the dependency graph are compatible and comprising default values for all input and output parameters;

mapping parameters of first and second functions of the first and second nodes, where the mapping comprises an index, defines a relationship where input parameters are ignored and output parameters are unmapped and take on default values, where parameter value and type are passed for the mapping and the function data structure and value index are passed for the mapping; and

evaluating the function as part of an evaluation of the second dependency node comprising determining a type of a passed parameter where parameter types are identified dynamically as the dependency graph is executed.

18. A method as recited in claim 17, wherein the mapping comprises an index remapping and a matrix of data casting methods which will change one type of data into another.

19. A method comprising:

passing a function from a first node in a node network to a second node in the node network; and

evaluating the function as part of an evaluation of the second node.

20. An apparatus comprising a computer including a dependency node evaluation system having functions passed between nodes of a dependency graph of a graphics creating process.

21. A data structure provided on computer readable storage controlling a computer in association with evaluating a dependency graph of a graphics creation process, the data structure comprising an RTTI

parameter list, a mapping substructure comprising an index mapping, mapping methods, and a data casting matrix, a function pointer, and methods for setting inputs, getting outputs, and evaluating a passed function.

22. A method of evaluating a dependency graph of a graphics creation process, comprising performing, by a destination node, of an algorithm having a function known to the destination node by evaluating a self evaluating data structure passed from a source node and expected to precisely implement the function known to the destination node where the self evaluating data structure can comprise a different function with different parameters and performing the different function actually requested by the destination node.